



Radar Systems and
Remote Sensing Laboratory

①

AD-A258 314



A NUMERICAL PROCEDURE FOR
RECOVERING TRUE SCATTERING COEFFICIENTS
FROM MEASUREMENTS WITH
WIDE-BEAM ANTENNAS

DTIC
ELECTE
DEC 2 1992
S B D

DISTRIBUTION STATEMENT A
Approved for public release
Distribution Unlimited

THE UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.

2291 Irving Hill Road
Lawrence, Kansas 66045-2969

**A NUMERICAL PROCEDURE FOR
RECOVERING TRUE SCATTERING COEFFICIENTS
FROM MEASUREMENTS WITH
WIDE-BEAM ANTENNAS**

Qinglin Wang and S. P. Gogineni

**Radar Systems and Remote Sensing Laboratory
University of Kansas Center for Research, Inc.
2291 Irving Hill Road
Lawrence, Kansas 66045-2969
913/864-4835
FAX: 913/864-7789 OMNET: KANSAS.U.RSL
TELEX: 706352**

RSL Technical Report 8240-2

September 1991

Sponsored by:

**NASA Headquarters
Washington DC 20546
Grant No. NAGW-1984
and
Office of Naval Research
Arlington VA 22217-5000
Grant No. N00014-89-J-1456**

92-30579



420 122 298

9 2 1

~~91 10 3 0 5 4~~

CONTENTS

1. Introduction.....	2
2. Correction of the Narrow-beam Approximation Error	5
3. Exponential Model: A Piecewise-Linear Approximation.....	6
4. Estimation of Model Parameter : Lookup Table Method	8
4.1 Calculated σ^0	8
4.2 Problem description.....	9
4.3 Lookup table method.....	10
4.4 Generation of lookup table	11
4.5 Parameter estimation	12
5. Estimation of Backscattering Coefficient σ^0	14
6. Results	15
7. Summary	16
8. References	17
9. Appendix Pascal Program For Generating Lookup Table	18

Accession For	
NTIS GRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
<i>per letter</i>	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
<i>A-1</i>	

RSL Technical Report

A Numerical Procedure for Recovering True Scattering Coefficients from Measurements with Wide-Beam Antennas

Qinglin Wang and Sivaprasad Gogineni

Radar Systems and Remote Sensing Laboratory
University of Kansas, Lawrence, KS 66045

Abstract This report presents a numerical procedure for estimating the true scattering coefficient, σ^0 , from measurements made using wide-beam antennas. The use of wide-beam antennas results in an inaccurate estimate of σ^0 if the narrow-beam approximation is used in the retrieval process for σ^0 . To reduce this error, we propose a correction procedure that estimates the error resulted from the narrow-beam approximation and uses the error to obtain a more accurate estimate of σ^0 . An exponential model has been assumed to take into account the variation of σ^0 with incidence angles, and the model parameters are estimated from measured data. Based on the model and knowledge of the antenna pattern, the procedure calculates the error due to the narrow-beam approximation. The procedure is shown to provide a significant improvement in estimation of σ^0 obtained with wide-beam antennas. The proposed procedure is also shown insensitive to the assumed σ^0 model.

1. Introduction

Scatterometers, calibrated radar systems, are designed to estimate, accurately and precisely, the backscattering coefficient σ^0 of homogeneous targets from the received power. Because the backscattered signal is the superposition of all scatterers illuminated by the antenna beam, the implicit assumptions of this retrieval procedure are that those illuminated scatterers have the same property as far as backscattering is concerned, and the radar system has the same response to all the backscattered components. Practically, these assumptions are approximately valid only when a narrow-beam antenna is used in the measurement, and common retrieval procedures are based on so-called narrow-beam approximation, which will be explained in following discussion.

For a monostatic radar, the received power from a distributed target is, mathematically, related to σ^0 by an integral equation, the radar equation, as expressed by eq. (1)

$$P_r(\theta) = \frac{P_t \lambda^2 G_0^2}{(4\pi)^3} \int_A \frac{g^2(\theta_a, \phi_a, \theta)}{R^4(\theta_a, \phi_a)} \sigma^0(\theta_a, \phi_a) dA \quad (1)$$

where

- P_t = transmitted power
- λ = wavelength
- G_0 = maximum gain of antenna
- g^2 = normalized two-way antenna gain function
- σ^0 = scattering coefficient of the target
- R = range between antenna and target
- θ = incidence angle of EM waves (see Fig. 1),

The radar illumination geometry is illustrated in Fig.1. Notice that σ^0 , g^2 , and R are the functions of radar coordinates (θ_a, ϕ_a) , and the integration is carried out over the illuminated area A . The received power P_r is a function of incidence angle θ , as indicated in (1).

Our objective is to retrieve $\sigma^0(\theta)$ from $P_r(\theta)$ based on (1), and assumptions are made to facilitate the retrieval procedure. One of these assumptions is the narrow-beam approximation, which states that, if the antenna beam is sufficiently narrow, σ^0 , g^2 , and R can be considered approximately as constants over the illuminated area A . Under the narrow-beam approximation, variables σ^0 , g^2 , and R in (1) are replaced by their representative values evaluated at incidence

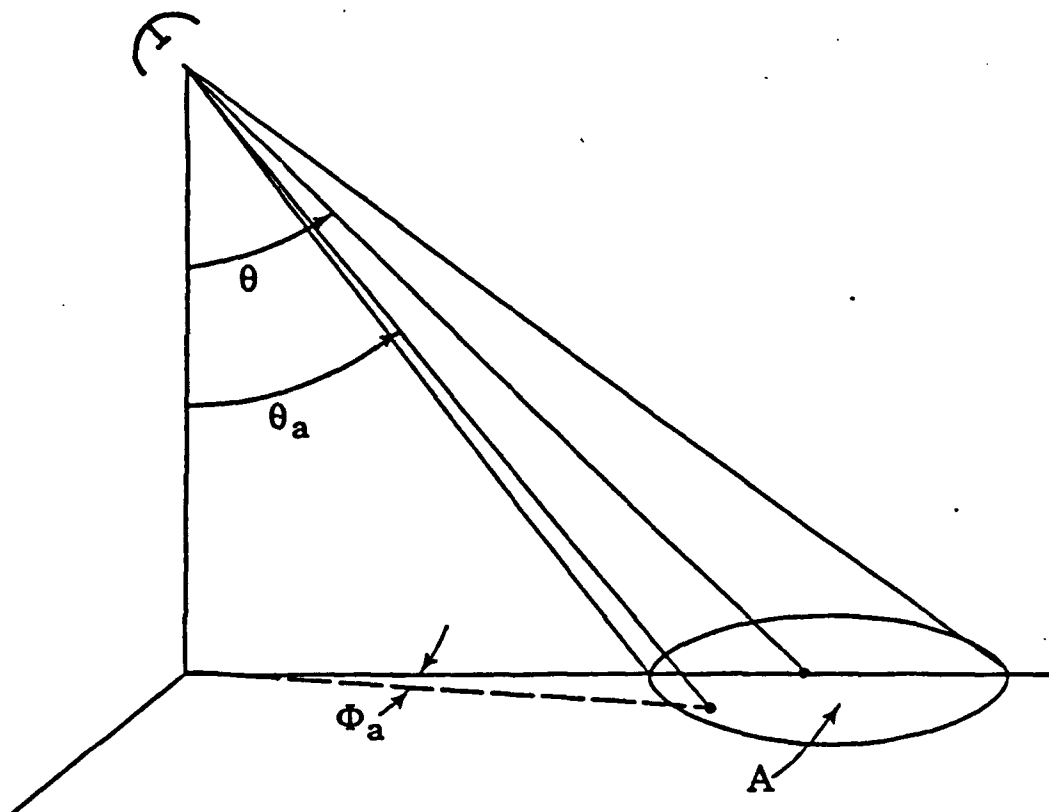


Fig. 1 Illumination geometry

angle θ , and (1) can be simplified as

$$P_r(\theta) = \frac{P_t \lambda^2 G_0^2 \sigma^0(\theta)}{(4\pi)^3 R^4(\theta)} A_{ill}(\theta) \quad (2)$$

where $A_{ill}(\theta) = \int_A dA$ is the "effectively" illuminated area. A common practice is to define $A_{ill}(\theta)$ as the area of the effective antenna beamwidth intercepted by the target plane. For a pencil beam, the intercepted area is an ellipse. Using (2), σ^0 can be retrieved from measured data, radar parameters and geometry,

$$\sigma_s^0(\theta) = \frac{P_r(\theta)(4\pi)^3 R^4(\theta)}{P_t \lambda^2 G_0^2 A_{ill}(\theta)} \quad (3)$$

Equation(3) gives so-called measured σ^0 (denoted as σ_s^0), which is calculated under the narrow-beam approximation. Using (1), (2) and (3), $\sigma_s^0(\theta)$ can be related to true $\sigma^0(\theta)$ as

$$\sigma_s^0(\theta) = \sigma^0(\theta) \frac{1}{A_{ill}(\theta)} \int_A \frac{g^2(\theta_a, \phi_a, \theta)}{R^4(\theta_a, \phi_a, \theta)} \sigma^0(\theta_a, \phi_a, \theta) dA \quad (4)$$

where

$$R(\theta_a, \phi_a, \theta) = \frac{R(\theta_a, \phi_a)}{R(\theta)}, \text{ normalized range}$$

$$\sigma^0(\theta_a, \phi_a, \theta) = \frac{\sigma^0(\theta_a, \phi_a)}{\sigma^0(\theta)}, \text{ normalized scattering coefficient}$$

$$g^2(\theta_a, \phi_a, \theta), \text{ normalized two-way antenna pattern as in (1).}$$

From (4) it can be seen that the σ_s^0 represents only an averaged value of the true σ^0 weighted by the normalized antenna pattern and range. However, the condition of the narrow-beam approximation ensures that the integrand of (4) is approximately equal to one over the integration area, and σ_s^0 , the weighted average of σ^0 , is close to the true σ^0 at incidence angle θ .

Unfortunately, wide-beam¹ antennas are often used in scatterometers for various reasons, and the

narrow-beam approximation is then not a valid assumption. If eq. (3) is still used in the retrieval process, σ^0 could deviate from the true σ^0 by a large value. Ulaby et al. [1, pp. 755] briefly discusses the effect of assuming constant σ^0 in the illumination integral. With a beamwidth of 10° , the narrow-beam approximation could result in an error of several dBs in estimating σ^0 if true σ^0 changes rapidly over the illuminated area.

Two examples of the effects of the narrow-beam approximation are shown in Fig.2. The theoretical values of σ^0 of a smooth sea-ice surface at several angles are taken from [2, pp. 1754], and a smoothly fitted curve is generated from those data and extended to the entire relevant range of incidence angles. The curve is used to calculate received power from (1) with a pencil-beam antenna of Gaussian pattern. The calculated power is then used to calculate σ^0 from (3), which is derived under the narrow-beam approximation. As a quantitative illustration, the figure shows how large the narrow-beam approximation error could be if a wide-beam antenna is used to measure σ^0 when it changes rapidly over the angular extent of the antenna beamwidth. Because the σ^0 of targets with a smooth surface often exercises a sharp change for small incidence angles, it is necessary to reduce the error to an acceptable extent if the narrow-beam approximation condition is violated in the measurement. In other words, some correction is needed on σ^0 to take into account the effect of average in (4), so that the requisite accuracy of estimation is achieved even though wide-beam antennas are used.

In section 2, we discuss the possible solutions and propose our approach. In sections 3-5, we describe the proposed procedure in detail. In section 6, we present two examples, and in section 7, we give a summary along with our conclusion.

-
1. In this report, an antenna is considered as "wide-beam" if the error in the estimate of σ^0 by eq. (3) is too large to be accepted. Otherwise, it is narrow-beam. These terms really involve three factors: antenna beamwidth, variation of σ^0 , and error tolerance in estimating σ^0 .

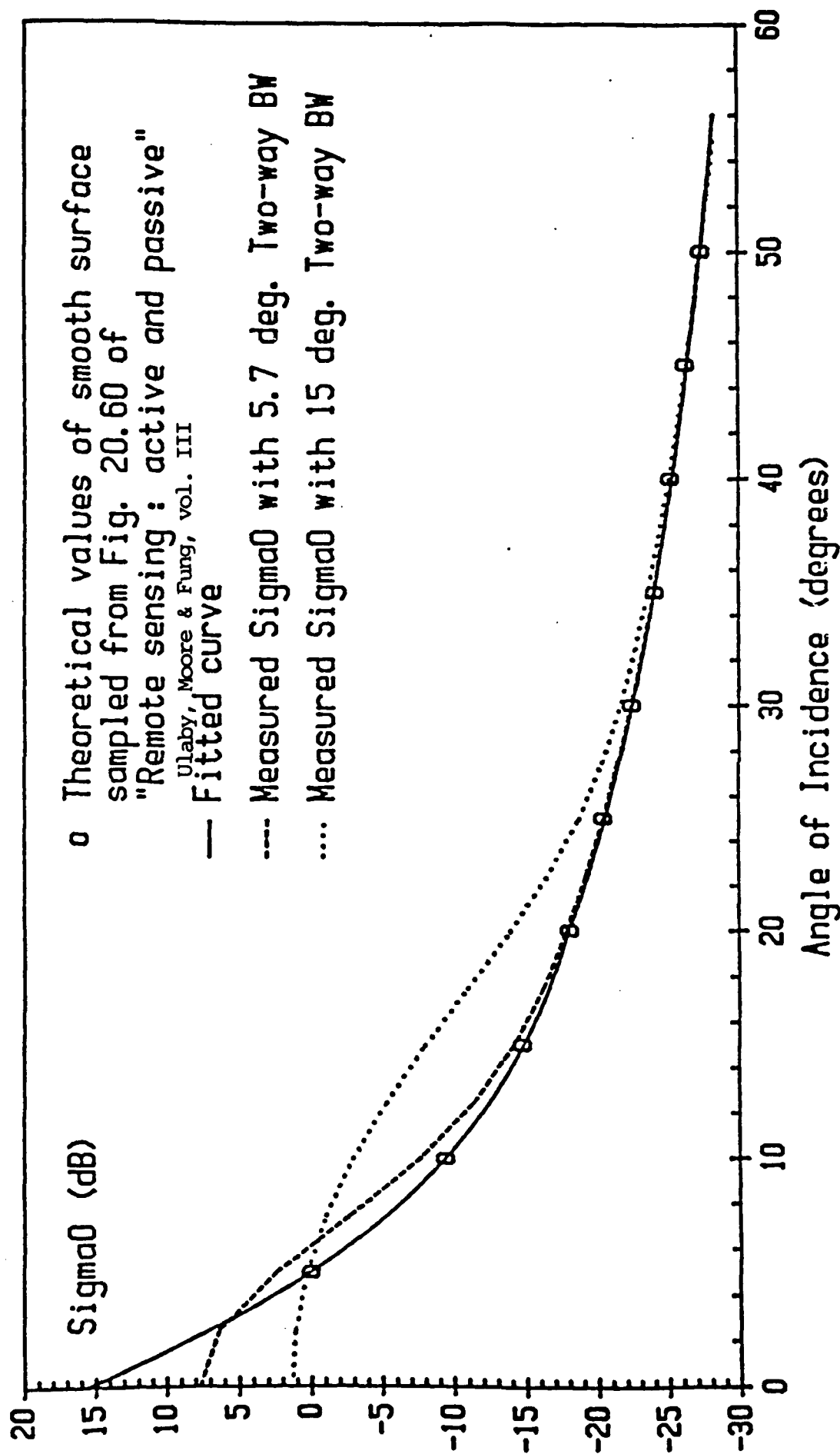


Fig. 2 Effects of narrow-beam approximation: smooth surface of sea ice

2. Correction of the Narrow-beam Approximation Error

Given received power $\{P_r(\theta_k)\}$ measured at incidence angles $\{\theta_k\}$, we try to estimate true $\sigma^0(\theta_k)$ even though the narrow-beam approximation may be violated. The difficulty of the problem is that, without the narrow-beam approximation, it involves an integral equation (see (1)). Observing (1), we can consider the function $g^2(\theta_a, \phi_a, \theta) / R^4(\theta_a, \phi_a)$ in the integrand as a two-dimensional spatial filter. The two-dimensional convolution, as represented by (1), states that the input to the filter is the true σ^0 , and the output of the filter is the power received by the antenna (notice that output is one-dimensional, or can be thought as having rotational symmetry). We want to find the input signal of the filter from the output, a deconvolution problem.

For a general two-dimensional deconvolution problem, an inverse filter can be structured to retrieve the input signal based on output data as well as characteristics of the filter and the measurement noise [3, pp. 61]. However, in our case the facts that the measurements are generally made over a small set of incidence angles and that the filter is space variant make the use of inverse filters questionable.

Axline [4] developed a matrix inversion procedure to invert the integral equation. One drawback of this method is that it needs $P_r(\theta_k)$ to be measured with small angular increments. If a theoretical or empirical model can be used to describe σ^0 , it is possible to develop algorithms to estimate the parameters of the assumed model [5][6].

Another approach for attacking the problem is, instead of solving the integral equation (1) directly, trying to improve σ_a^0 , the estimate based on the narrow-beam approximation, by adding a correction term, in terms of dB, which represents the error due to the narrow-beam approximation in the retrieval process. In the following discussion, we call the error term a narrow-beam approximation error, and name this kind of approach a narrow-beam approximation plus correction. The narrow-beam approximation error can be mathematically defined as

$$E(\theta_k) = \sigma^0(\theta_k) - \sigma_a^0(\theta_k) \quad (5)$$

A more accurate estimate of σ^0 can be obtained by adding this error term to σ_a^0 ,

$$\sigma^0(\theta_k) = \sigma_s^0(\theta_k) + \hat{E}(\theta_k) \quad (6)$$

where $\hat{E}(\theta_k)$ is the estimate of $E(\theta_k)$ and all terms in (5) and (6) are in dB.

$E(\theta_k)$ is unknown and eq. (5) cannot be used to calculate $E(\theta_k)$ because $\sigma^0(\theta_k)$ is unknown. However we can assume that true σ^0 belongs to (or is close to) a certain model and the model parameters can be estimated from measured data. Once the model is determined, the narrow-beam approximation error could be calculated via (4). The idea behind this approach is that what we need is the error $\hat{E}(\theta_k)$, the difference between $\sigma_s^0(\theta_k)$ and $\sigma^0(\theta_k)$ in (4) due to the effect of wide antenna beam, not the model itself, so an accurate model is not crucial as long as the error can be estimated from the model with accepted accuracy. In this report, we propose a numerical procedure to estimate the narrow-beam approximation error. Later we will see that the proposed procedure is indeed insensitive to assumed model.

The approach of narrow-beam approximation plus correction has been used by several researchers, e.g., Kim et al. [6] and Stiles et al. [7]. Kim et al. proposed an exponential model and a recursive algorithm to estimate the model parameters. Stiles et al. described an empirical formula, based on a group of hypothetical theoretical σ^0 curves, to calculate the error $E(\theta_k)$ for small incidence angles. The proposed approach in this report is more general and systematic.

3. Exponential Model: A Piecewise-Linear Approximation

In order to estimate the narrow-beam approximation error via (4), a parameterized σ^0 model as the function of incidence angle must be chosen. The choice of the model is based on the following considerations:

- a. It fits experimental data, at least approximately.
- b. It facilitates mathematical handling.
- c. It is robust; that is, deviation of true σ^0 from assumed model should not cause unacceptable error.

Note that the purpose of introducing the model is to estimate the error resulting from the narrow-beam approximation, so it is possible to obtain a good estimate of the error even though the

model itself may not accurately describe the true scattering characteristics of the target.²

A reasonable model is the piecewise-exponential model

$$\sigma^0(\theta) = A_i \exp(-\theta / B_i) \quad \theta_{L_{i-1}} \leq \theta < \theta_{L_i}, \quad i = 1, \dots, N \quad (7)$$

Generally $N = 2$, and the interval $(\theta_{L_{i-1}}, \theta_{L_i})$ can be identified by inspecting σ^0 or "optimum fitting."

There are two reasons for choosing the exponential model. First, the exponential model has been found to fit reasonably well experimental data from North American agricultural terrain, sea ice, and ocean[1, pp. 577]. Second, the model is simple and the model parameters are easily estimated from measured data. Particularly, the two model parameters, A_i and B_i , are separable so that a combined approach of a one-dimensional lookup table plus one-dimensional optimization can be established to estimate the parameters, as proposed in this report. Mathematically, the exponential model is nothing but a piecewise-linear approximation of the true σ^0 in the log scale (dB). Therefore, we can expect a better model by dividing the covered range into more segments, and each segment is modeled with distinct exponential parameters. As mentioned earlier, what we are concerned with is the narrow-beam approximation error rather than the model itself, and we will give an example to show that even if the true σ^0 deviates from the model, the results of the procedure are still reasonably good.

As indicated in (7), the so-called exponential model here is actually a piecewise-exponential model, or piecewise-linear model in log scale; that is, the model consists of several segments and each segment is modeled by a distinct exponential function. For most measured data, at least two segments are needed, one covering small incidence angles and another for large angles. Here the model is in contrast to the one used by Kim et al., who used one exponential function to cover all the relevant incidence angles.

2. The narrow-beam approximation error is defined in eq. (5), and the estimated error from the model is given by eq. (18). Any modeling error will result in errors in $\sigma_{mc}^0(\theta_k)$ (estimate of $\sigma^0(\theta_k)$) and in $\sigma_+^0(\theta_k)$ (estimate of $\sigma^0(\theta_k)$). However if the errors are positive correlated, they may cancel each other in the estimate of the narrow-beam approximation error given by eq. (18)

4. Estimation of Model Parameter : Lookup Table Method

4.1 Calculated σ^0

With the chosen model, denoted as σ_m^0 , the so-called calculated σ^0 (denoted as σ_+^0) can be derived as (see (4))

$$\sigma_+^0(\theta_k) = \frac{R^4(\theta_k)}{A_{ill}(\theta_k)} \int_A \frac{g^2(\theta_a, \phi_a, \theta)}{R^4(\theta_a, \phi_a)} \sigma_m^0(\theta_a, \phi_a) dA \quad (8)$$

where $A_{ill}(\theta_k)$ is the equivalent area as in (2).

As indicated by its name, σ_+^0 is obtained by numerically solving eq. (8). On the other hand, σ_-^0 is calculated from measured P_r based on eq. (3). The integration in eq. (4), which relates σ_-^0 to true σ^0 , is really embedded in the physical process of the measurement although (4) and (8) have exactly the same form. Another difference between (4) and (8) is that σ^0 in (4) is associated with the measured target and is unknown, and σ_m^0 is the parameterized model that we are trying to fit to measured data in the hope that narrow-beam approximation errors regarding σ_-^0 and σ_+^0 are close to each other. It is very clear that all other quantities appearing in (8), including antenna pattern and range, should represent the same quantities involved in the measurement process. Though errors are inevitable in modeling these quantities, we ignore these errors in following discussions. In other words, values of σ_-^0 are experimental data; and σ_+^0 , on the other hand, is only a mathematical expression that reflects the effect of the illumination integral on σ_m^0 .

To avoid confusion, some words are needed here about σ^0 , σ_m^0 , σ_-^0 , and σ_+^0 . σ^0 is the true backscattering coefficient, which is never known and which we are trying to estimate. σ_-^0 is the measured backscattering coefficient based on received power and the narrow-beam approximation (see (3)); it actually includes all effects of noise and errors in the process of measurement and calibration. σ_m^0 is the estimate of σ^0 based on the assumed model and estimated parameters. The last one, σ_+^0 , the calculated backscattering coefficient, is given by (8), and it reflects the illumination integral process in (1), which is ignored by the narrow-beam approximation.

One interesting point is that, even though we are trying to find the best fit of measured data within the chosen model, we do not attempt to use σ_m^0 as a final estimate of σ^0 , but only use it to find the narrow-beam approximation error. This is because we are not sure whether or not the model (and/or the estimated parameters) is accurate. However we believe that the error estimate obtained using the model is reasonably accurate for applying the correction procedure. Fig.3 gives a concise illustration of the above discussion. Next, we discuss the parameter estimation procedure in detail.

4.2 Problem description

Comparing (4) with (8), we can see that σ^0 will equal σ_+^0 if everything is a perfect match, particularly between the parameterized model σ_m^0 in (8) and true σ^0 , and all other error sources are ignored. Therefore, the difference between them can be used to assess the correctness of the parameter estimation. The problem can be mathematically described as follows:

Given a measured vector $\sigma^0 = (\sigma^0(\theta_1), \dots, \sigma^0(\theta_n))'$, where superscript t means the transposed vector, over a set of incidence angles $\{\theta_j\}$, we need to find the optimum model parameters A_0 and B_0 that minimize the squared Euclidean distance between the two vectors, σ^0 and σ_+^0 , defined by

$$\begin{aligned} D^2 &= \|\sigma^0 - \sigma_+^0\|^2 \\ &= \sum_{i=1}^n (\sigma^0(\theta_i) - \sigma_+^0(\theta_i))^2 \end{aligned} \tag{9}$$

where σ_+^0 is the so-called calculated vector

$$\sigma_+^0 = (\sigma_+^0(\theta_1), \dots, \sigma_+^0(\theta_n))'$$

whose components $\{\sigma_+^0(\theta_k)\}$ are obtained from (8) with

$$\sigma_m^0(\theta) = A \exp(-\theta / B)$$

In (9), the error is calculated in dB because σ^0 changes drastically with incidence angles. In addition, some kind of weighting can be incorporated into the expression to emphasize part of the measured data.

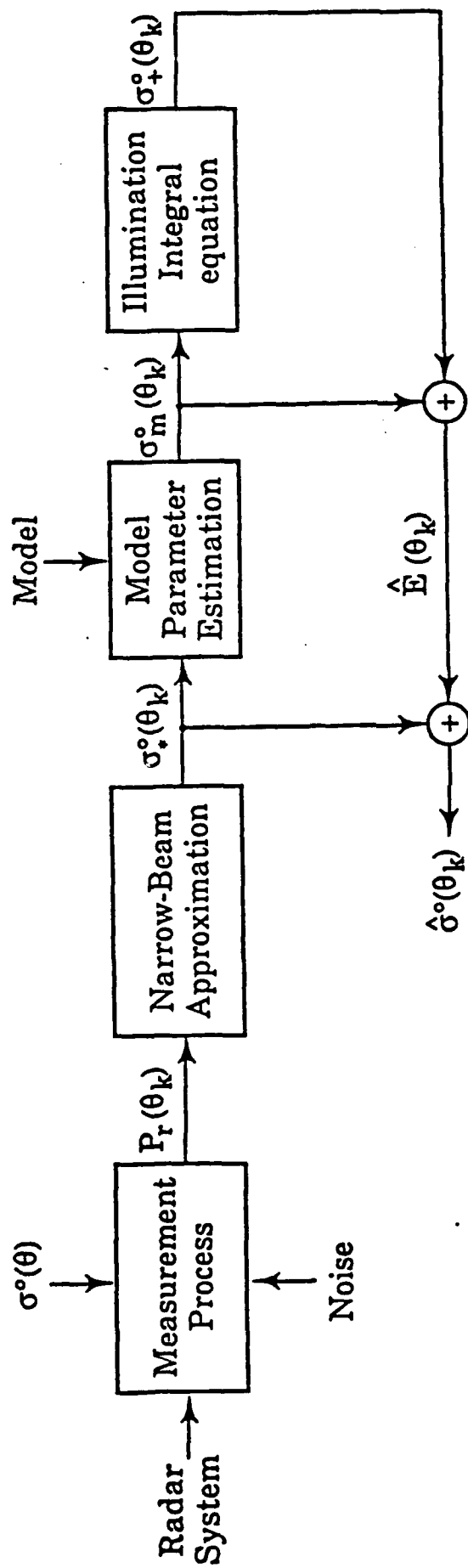


Fig. 3 Block diagram of correction procedure

4.3 Lookup table method

Various two-dimensional minimum-finding procedures can be used to find the minimum solution of (9), but three factors must be taken into account:

- a. Does the procedure converge each time?
- b. Does it converge to the global minimum each time?
- c. How fast does it converge?

It is not always easy to answer these questions. One way to get around the above difficulties is the lookup table method. That is, calculate $\{\sigma_+^0(\theta_i)\}$ over possible ranges of parameters A and B and store the results in a table in advance. Then in the retrieval process, the optimum parameters of the model can be obtained by simply comparing measured data σ^0 with each entry of the table and finding the one that has the smallest distance. Obviously the lookup table method can only yield a sub-optimum solution in the sense that the parameters A and B must be quantized into finite sets so that the minimum solution is "optimum" only as related to the chosen parameter sets. The quantized error can be reduced by using a finer quantization step, or a larger-size table. On the other hand, we have assumed that the proposed approach is not model-sensitive and we are not claiming the exponential model always accurately represents the true σ^0 . That means the narrow-beam approximation error cannot be very sensitive to the model parameters either, so that the parameters A and B do not need to be quantized with very small steps. Otherwise, the whole procedure is questionable. The incidence angle θ must also be quantized into a finite set, which is less problematic because measurements are generally taken over a set of pre-designed incidence angles. Furthermore, interpolation techniques can be used if occasionally the measured data have different angles from those in the table. Another problem is the memory requirement for storing the table. Totally there are $N_\theta \times N_A \times N_B$ values in the table where N_θ , N_A , and N_B are the sizes of quantized θ , A, and B respectively. This requirement may impose serious problems in practice.

One reason for us to choose the exponential model is its mathematical advantage. In particular, the two parameters of the model are separable, and the "best" parameter A is easy to obtain if B is fixed. As a result, instead of a two-dimensional table, only a one-dimensional table with B as parameter is needed for our model fitting, as seen soon.

Observe that in equation (8), parameter A of the exponential model only acts as a scale factor. Therefore, we can separate the effects of A and B by expressing $\sigma_+^0(\theta_k)$ as $A \sigma_-^0(\theta_k)$ where $\sigma_-^0(\theta_k)$ is still calculated from (8) but instead of using the model in (7), the normalized model

$$\sigma_n^0(\theta) = \exp(-\theta / B) \quad (10)$$

is used in the expression.

With the above notations, the problem can be restated as

$$\begin{aligned} \min_{A,B} D^2 &= \min_{A,B} \|\sigma_-^0 - \sigma_+^0\|^2 \\ &= \min_B (\min_A \|\sigma_-^0 - \sigma_+^0\|^2) \\ &= \min_B (\min_A \|\sigma_-^0 - A \sigma_-^0\|^2) \end{aligned} \quad (11)$$

The significance of (11) is that the two-dimensional search has been translated to two one-dimensional searches. This makes it suitable to use only a one-dimensional lookup table to store σ_-^0 , which is only the function of the model parameter B, and to estimate A directly in the model-fitting process. Therefore we not only drastically reduce the storage requirement of the lookup table method by a factor N_A , the size of quantized A, but also eliminate the quantization error of A because now it is estimated over a continuous base.

4.4 Generation of lookup table

Each element of the lookup table consists of a particular value of B, say B_i out of the set of quantized $\{B_k\}$, and two corresponding vectors, $\sigma_{-,i}^0$ and E_i .

The vector $\sigma_{-,i}^0$ is defined as

$$\sigma_{-,i}^0 = (\sigma_{-,i}^0(\theta_1), \dots, \sigma_{-,i}^0(\theta_n))' \quad (12)$$

where $\sigma_{-,i}^0(\theta_k)$ is calculated from (8) using the normalized model in (10) with parameter B_i .

E_i is the error vector whose the k^{th} element, $e_i(\theta_k)$, is simply the difference between $\sigma_{-,i}^0(\theta_k)$ and $\exp(-\theta_k / B_i)$, the value of the model at θ_k , both in dB. This error vector will be used for the error correction.

How to choose the quantized parameters set $\{B_i, i = 1, 2, \dots, M\}$ depends on the allowable memory size, the acceptable time spent on the calculation, and the tolerable error due to quantization on B. The quantization errors can be assessed by observing the difference between the components of two vectors, $\sigma_{-,i}^0$ and $\sigma_{-,i+1}^0$, corresponding to two adjacent quantized values, B_i and B_{i+1} . The set of $\{\theta_i, i = 1 \dots n\}$ is chosen to include all possible incidence angles, in consistence with the designed experiment.

For each quantized B_i , $\sigma_{-,i}^0(\theta_k)$ is found over the chosen incidence angles $\{\theta_j\}$ by numerically solving the integral (8). Besides the normalized exponential model (10) with specified B_i , the normalized two-way antenna pattern $g^2(\cdot)$ must be specified in order to carry out the integral, which could be a fitted function of measured antenna pattern or just measured antenna data if the measurements were made over fine-enough angles. For the latter case, interpolation might be needed in the calculation. The chosen integral area A should be large enough so that the contribution to the integral from the outside of the area can be ignored. Note that only the relative distance $R(\theta_a, \phi_a) / R(\theta_k)$ appears in (8), so the real height of the antenna position is irrelevant and any height can be assumed for the calculation.

A Pascal program for generating a required lookup table is listed in the Appendix. The integration program used was taken from [8] and slightly modified for two-dimensional integration. In the program, the two-way antenna gain was assumed to be a Gaussian pattern.

Generating the lookup table is a time-consuming task. As a matter of fact, the integration (8) needs to be carried out $N_\theta \times N_B$ times where N_θ and N_B are, respectively, the sizes of quantized θ and B. However, the table has been made once and for all as long as the same antenna is used. The model fitting costs little time once the table has been made.

4.5 Parameter estimation

The parameter estimation procedure is straightforward once the lookup table is made. Fig.4 diagrammatically illustrates the procedure. The squared distance between σ_θ^0 and σ_B^0 is plotted in the form of equi-distance contours with parameters A and B as independent variables. Parameter

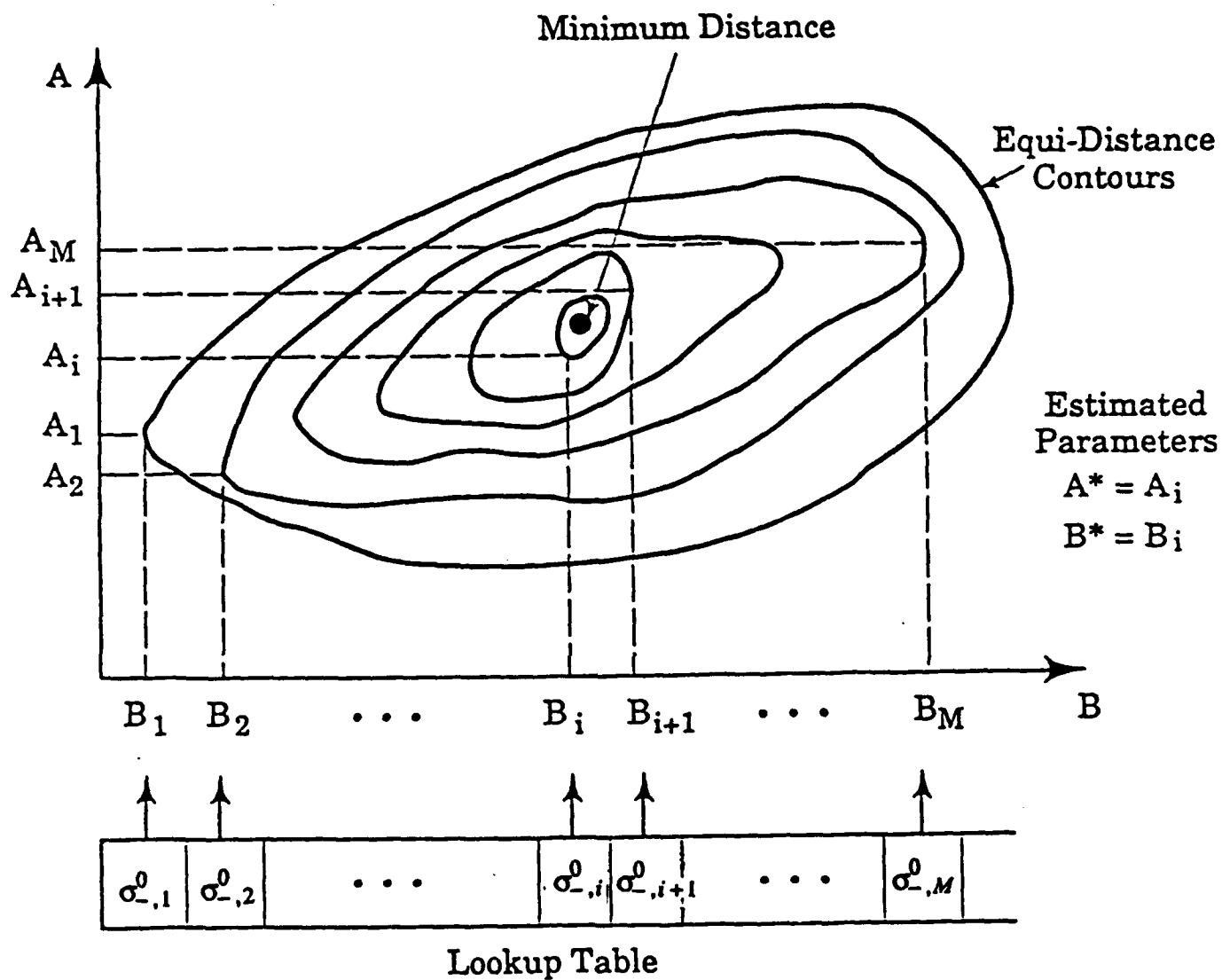


Fig. 4 Parameter searching

B has been quantized into a set of discrete values $\{B_i, i=1, \dots, M\}$ and a lookup table has been made whose i th element is a vector σ_{-i}^0 given in (12). For each discrete B_i (equivalently, σ_{-i}^0 , the element of the lookup table), the local minimum distance D_i^2 is calculated with the fixed B_i and the free parameter A. By comparing $\{D_i^2\}$, the model parameters are declared as the ones that yield the smallest distance D^* among $\{D_i^2\}$. As seen from the figure, quantization on parameter B results in some error in locating the global minimum. The error can be made as small as we wish by reducing the quantization step with the price of a larger table and longer searching time. Also two-dimensional interpolation techniques can be used to reduce the error further. The step-by-step procedures for estimating the model parameters are as follows:

Step 1 Calculate σ_*^0 vector from measured data using (3).

Step 2 For each element of the table, σ_{-i}^0 vector, find optimum A_i , which minimizes the distance between the measured vector σ_*^0 and $\sigma_{+i}^0 = A\sigma_{-i}^0$, according to the following equation

$$A_i = \frac{1}{K} \sum_{j=1}^K (\sigma_*^0 \theta_j) - \sigma_{-i}^0(\theta_j) \quad (dB) \quad (13)$$

which can be proven easily, as shown in following analysis.

For a fixed B_i , we want to find the optimum parameter A_i that minimizes the squared distance

$$\begin{aligned} D_i^2 &= \|\sigma_*^0 - A_i \sigma_{-i}^0\|^2 \\ &= \min_A \|\sigma_*^0 - A \sigma_{-i}^0\|^2 \end{aligned} \quad (14)$$

Notice that in the above equation, σ^0 s are in dB as mentioned earlier, i.e.,

$$(A \sigma_{-i}^0)_{dB} = (A)_{dB} + (\sigma_{-i}^0)_{dB}$$

so

$$D_i^2 = \min_A \|\sigma_*^0(dB) - A(dB) - \sigma_{-i}^0(dB)\|^2 \quad (15)$$

$$= \min_A \sum_{j=1}^K (\sigma_j^0(\theta_j) - A - \sigma_{-i}^0(\theta_j))^2 \quad (dB)$$

Differentiate (15) and let the result equal zero, we get (13).

Step 3 Compare $\{D_i^2\}$ to find the smallest one

$$D^* = \min_i \{D_i^2\} \quad (16)$$

and the corresponding parameters A^* and B^* , which yield D^* , are the fitted parameters of the measured vector σ_j^0 based on quantized parameters $\{B_i\}$. The error vector corresponding to B^* can be read out from the table for error correction.

5. Estimation of Backscattering Coefficient σ^0

Generally, the angle response of σ^0 can be divided into two regions (see (7)), by either inspecting $\{\sigma_j^0(\theta_j)\}$ or trying different segmentations and finding a suitable division that yields minimum combined D^* . After the model parameters for each region are determined, the corresponding error vectors can be used to correct the narrow-beam approximation errors on the measured data $\{\sigma_j^0(\theta_j)\}$. However, the correction at those incidence angles near the intersection of the two fitted model segments might not be as accurate as for other angles. A better procedure is to use the combined model to directly calculate the error vector. With the fitted parameters, the combined model is

$$\sigma_{mc}^0(\theta) = \begin{cases} A_1^* \exp(-\theta / B_1^*) & 0 \leq \theta < \theta_0 \\ A_2^* \exp(-\theta / B_2^*) & \theta_0 \leq \theta < \pi/2 \end{cases} \quad (17)$$

The combined model is again used in (8) to calculate the error of the narrow-beam approximation, $E(\theta_k)$

$$E(\theta_k) = \sigma_{mc}^0(\theta_k) - \sigma_+^0(\theta_k) \text{ (dB)} \quad (18)$$

where $\sigma_+^0(\theta_k)$ comes from (8) using the combined model (17). Because $E(\theta_k)$ reflects the effect of the illumination integral (1), we can expect a better estimate of σ^0 to be

$$\delta^0(\theta_k) = \sigma_+^0(\theta_k) + E(\theta_k) \text{ (dB)} \quad (19)$$

We call δ^0 as corrected σ^0 because unlike σ_+^0 (uncorrected), a correction factor, $E(\theta)$, has been added to overcome the error of the narrow-beam approximation.

6. Results

The proposed procedure has been tested on different types of data, and in this section two examples are given to show the performance of the procedure. In the first example, the procedure was used on data obtained by sampling the curves labeled "measured σ^0 " in Fig.2 from 0° to 50° with steps of 2.5° . The corrected σ^0 s ($\{\delta^0(\theta_k)\}$), the measured σ^0 s ($\{\sigma_+^0(\theta_k)\}$), as well as the true σ^0 s, are plotted in Fig.5. The figure clearly shows the improvement in the estimation of σ^0 using the procedure described. Notice that the two-segment exponential model is not a suitable model for the smooth surface σ^0 curve, as shown in the Fig.2. The remarkable improvement justifies the earlier statement that the procedure is not model sensitive.

Another example uses true experimental data that were collected by two different systems.³ The measured σ^0 (σ_+^0) of both systems and the corrected σ^0 (δ^0) for the wide-beam antenna data are plotted in Fig.6. No correction was made on the narrow-beam antenna data because, except at

3. Two radar systems were used to record data simultaneously during the CRREL 1989 experiment. One radar was a FM-CW system with an antenna of 3° beamwidth, and another was a step-frequency system with an antenna of 15° beamwidth.

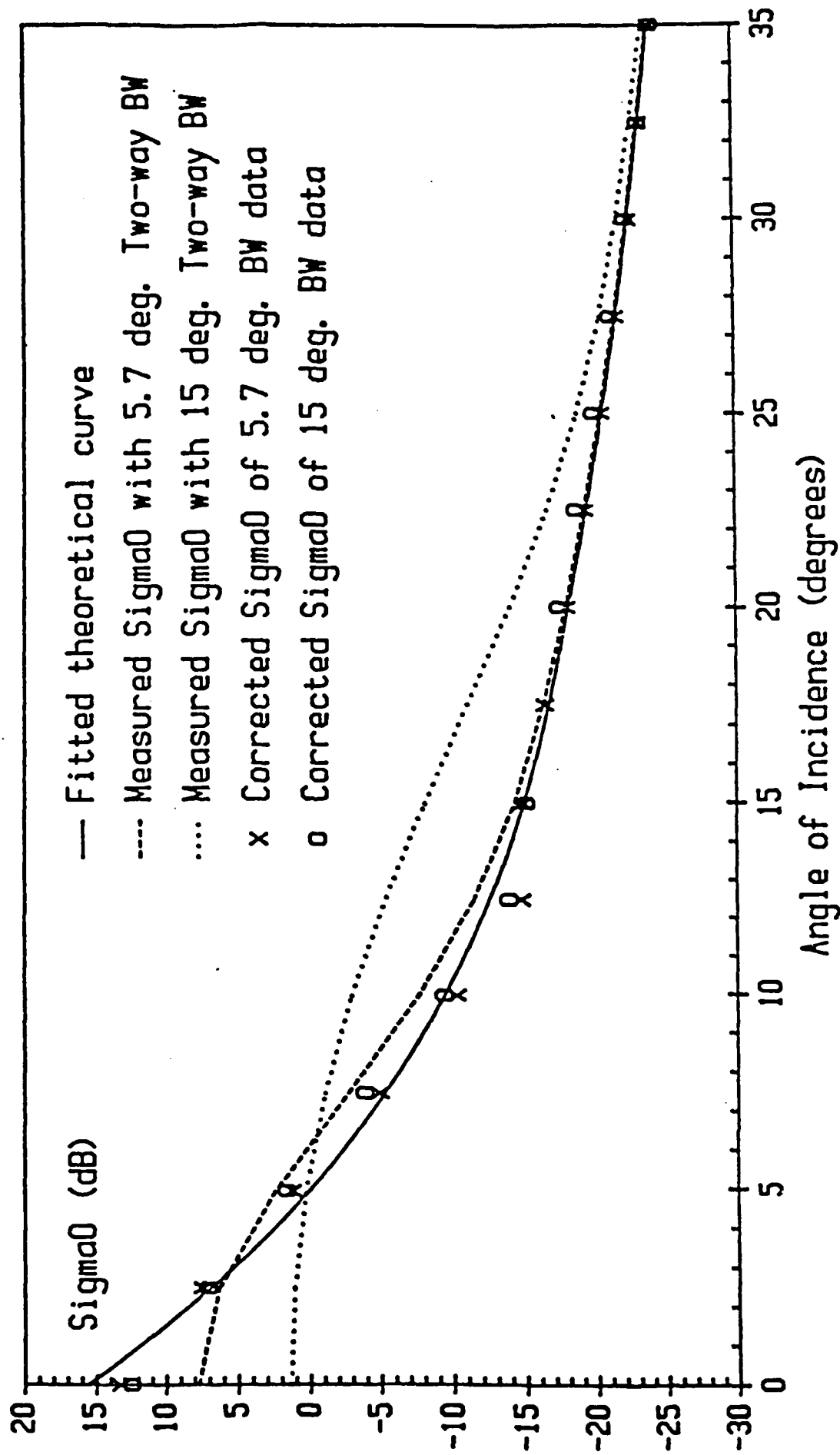


Fig. 5 Performance of correction: smooth surface of sea ice

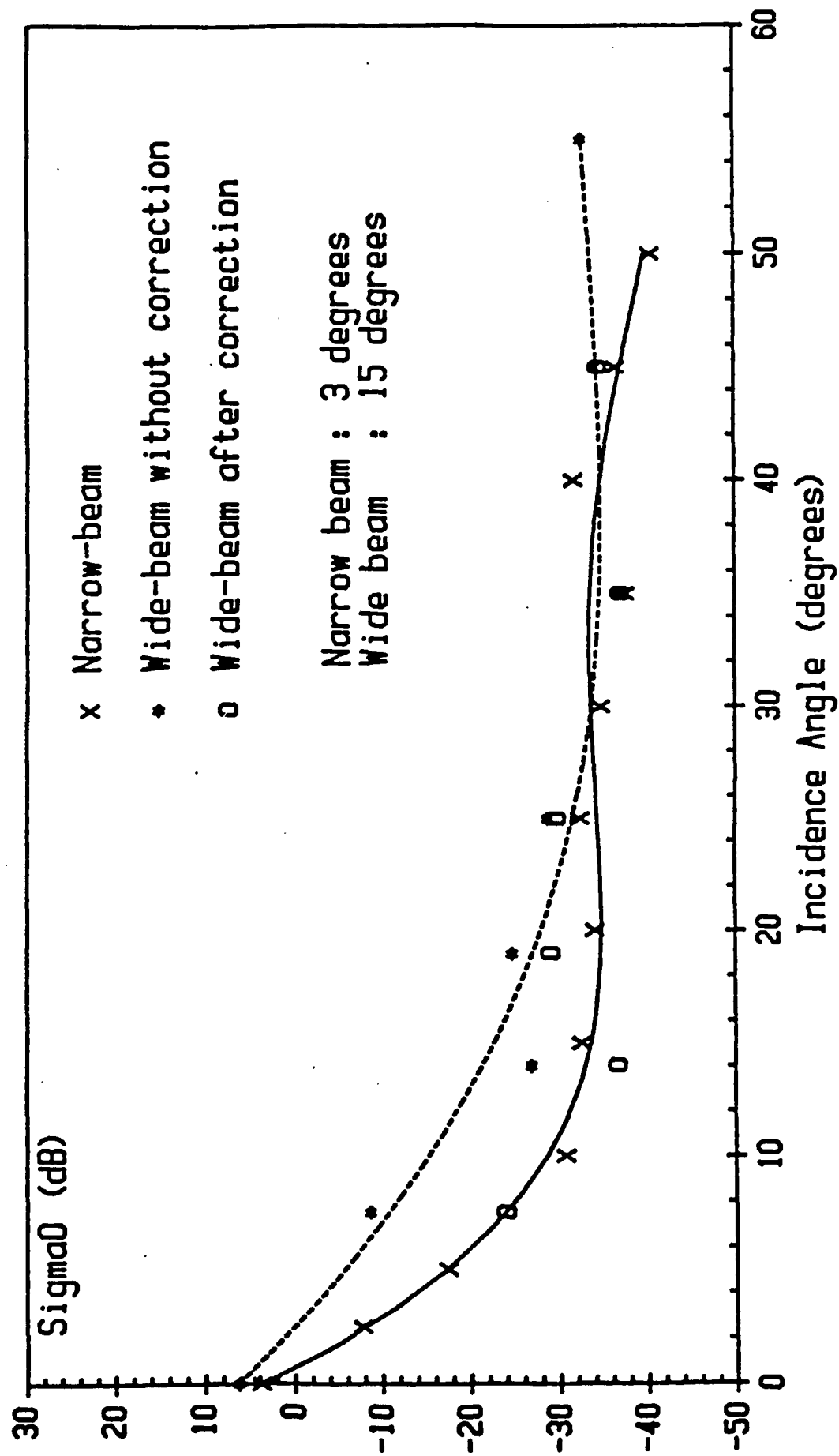


Fig. 6 Comparison of narrow and wide beam data (C band, 1-17-89)

normal incidence, the narrow-beam approximation errors were very small and negligible. The figure shows that, as expected, the wide-beam antenna system only gives an erroneous estimate of σ^0 for small incidence angles if the narrow-beam approximation is used to retrieve σ^0 . On the other hand, data from both systems show good consistency after the correction was made on the wide-beam data.

7. Summary

In this report, we have presented a correction procedure for the estimation of σ^0 for data collected by a wide-beam system. The procedure estimates the error incurred by the narrow-beam approximation and provides a more accurate estimate of σ^0 by correcting the error in some degree. An exponential model is used in the procedure, and a combined approach of a one-dimensional lookup table plus one-dimensional optimization is proposed for model fitting. The procedure is shown to provide significant improvement on the estimate of σ^0 and to be insensitive to the model assumption of σ^0 .

8. References

- [1] F.T.Ulaby, R.K.Moore, and A.K.Fung, *Microwave Remote Sensing: Active and Passive*, vol. II, Norwood, MA: Artech House, 1982.
- [2] F.T.Ulaby, R.K.Moore, and A.K.Fung, *Microwave Remote Sensing: Active and Passive*, vol. III, Norwood, MA: Artech House, 1986.
- [3] B.R.Hunt, "Image restoration," in *Digital Image Processing Techniques*, M.P.Ekstrom, ed., Orlando, FL: Academic Press, Inc., 1984.
- [4] R.M.Axline, Jr., "Experimental and simulated study of scattering from randomly rough surfaces," Univ. of Kansas Center for Res., Inc., Lawrence, RSL Tech. Report 222-2 (also Ph.D thesis, Univ. of Kansas, Lawrence), 1974.
- [5] F.T.Ulaby et al., "Method for retrieving the true backscattering coefficient from measurements with a real antenna," *IEEE Trans. Geosci. Remote Sensing*, vol. 21, pp.308-313, 1983.
- [6] Y.S.Kim et al., "Scattering coefficient estimate: an examination of the narrow-beam approximation," Univ. of Kansas Center for Res., Inc., Lawrence, RSL Tech. Report 331-23, 1982.
- [7] W.H.Stiles et al., "Performance analysis of the MAS (microwave active spectrometer) systems: calibration, precision, and accuracy," Univ. of Kansas Center for Res., Inc., Lawrence, RSL Tech. Report 360-4, 1979.
- [8] W.H.Press et al., *Numerical Recipes In Pascal: The Art of Scientific Computing*, New York: Cambridge Univ. Press, 1989.

9. Appendix Pascal Program For Generating Lookup Table

```
PROGRAM BEAMINT;
```

```
CONST
```

```
PI = 3.1415926536;  
HEIGHT = 1;  
H2 = 1;          { H2 = H*H }  
D_R = 0.0174532925;  
BEAMWIDTH_3DB = 0.2617994; { 15 DEGREES 5 GHZ }  
{ BEAMWIDTH_3DB = 0.0994838; 5.7 DEGREES 9.6 GHZ }  
{ BEAMWIDTH_3DB = 0.0877028; 5.025 DEGREES 13.6 GHZ }  
S_THETA = 2.5; { DEGREE }
```

```
TYPE
```

```
GLNARRAY = ARRAY[1..10] OF DOUBLE;
```

```
VAR
```

```
THETA,HALFBEAM_INT,CENTER_Y,LONG_AXIS,SHORT_AXIS,X1,Y1,Y2,Y0 : DOUBLE;  
RHO1,RHO2,PHI1,SINT,S1_CIRCLE,S2_CIRCLE,R,R0,X,Y : DOUBLE;  
R_THETA,PAR,S_PAR,R4,SIGMA,HALFBEAM_EFF,AILL : DOUBLE;  
CIRCLE,I,J : INTEGER;  
FDAT : TEXT;
```

```
FUNCTION DB(X:DOUBLE):DOUBLE;
```

```
BEGIN
```

```
DB := 10*LN(X)/LN(10)  
END;
```

```
FUNCTION TAN(X:DOUBLE):DOUBLE;
```

```
VAR
```

```
C : DOUBLE;
```

```
BEGIN
```

```
C := COS(X);  
IF (C = 0) THEN TAN := 1E36  
ELSE TAN := SIN(X)/C  
END;
```

```
{ arccos function }
```

```
FUNCTION ACOS(C:DOUBLE):DOUBLE;
```

```
VAR
```

```
S,AC : DOUBLE;
```

```

BEGIN
  IF (ABS(C)>1) THEN BEGIN
    IF (C>0) THEN C:= 1
      ELSE C:=-1
  END;
  IF (C=0) THEN
    AC := PI/2
  ELSE
    BEGIN
      S := SQRT(1.0-SQR(C));
      AC := ARCTAN(S/C);
      IF (AC < 0) THEN AC := PI+AC
    END;
    ACOS := AC
  END;

  { antenna gain function }

  FUNCTION ANTENNA_PATTERN(PHI : DOUBLE):DOUBLE;
  CONST
    C = 1.6651092;

  BEGIN
    ANTENNA_PATTERN := EXP(-SQR(C*PHI/BEAMWIDTH_3DB))
  END;

  { sigma0 function : exponential model }

  FUNCTION SIGMA0(PHI,PAR : DOUBLE):DOUBLE ;
  BEGIN
    SIGMA0 := EXP(-PHI/PAR)
  END;

  { find angle between the antenna look_direction and ground a pixel }

  FUNCTION ANGLE(RHO,PHI,R,Y0,R0 : DOUBLE):DOUBLE;
  VAR
    CPHI :DOUBLE;

  BEGIN
    CPHI := (RHO*COS(PHI)*Y0+H2)/(R*R0);
    ANGLE := ACOS(CPHI)
  END;

  { ANTENNA GAIN function :g**2 }

  FUNCTION FUNC(PHI,RHO:DOUBLE):DOUBLE;
  VAR

```

```

ANG :DOUBLE;
BEGIN
  ANG := ANGLE(RHO,PHI,R,Y0,R0);
  FUNC := ANTENNA_PATTERN(ANG);
END;

```

{ find parameters of illumination ellipse }

```

PROCEDURE ELLIPSE(THETA,HALFBEAM_INT : DOUBLE ;
  VAR CENTER_Y, LONG_AXIS, SHORT_AXIS : DOUBLE);
VAR
  A,B,C,D,E : DOUBLE;
BEGIN
  A := SQR(TAN(HALFBEAM_INT));
  B := SQR(COS(THETA))-A*SQR(SIN(THETA));
  C := HEIGHT*(1+A)*SIN(2*THETA);
  D := SQR(HEIGHT)*(A*SQR(COS(THETA))-SQR(SIN(THETA)));
  CENTER_Y := C/(2*B);
  E := D+SQR(C)/(4*B);
  LONG_AXIS := SQR(E/B);
  SHORT_AXIS := SQR(E)
END;

```

{ find angle of a ellipse given range in polar coordinate system }

```

FUNCTION PHI_LIMIT(CENTER_Y, LONG_AXIS, SHORT_AXIS, RHO : DOUBLE):DOUBLE;
CONST
  NOT_ZERO = 1E-30;
VAR
  A,B,C,D,Y_COF,X_COF,LIMIT : DOUBLE;
BEGIN
  Y_COF := SQR(LONG_AXIS);
  X_COF := SQR(SHORT_AXIS);
  A := SQR(RHO)*(Y_COF-X_COF);
  B := 2*X_COF*CENTER_Y*RHO;
  C := X_COF*Y_COF-Y_COF*SQR(RHO)-X_COF*SQR(CENTER_Y);
  D := (-B+SQR(SQR(B)-4*A*C))/(2*A);
  LIMIT := ACOS(D);
  IF (LIMIT = PHI1) THEN LIMIT := PHI1 + NOT_ZERO;
  PHI_LIMIT := LIMIT;
END;

```

{polynomial extrapolation function }

```

PROCEDURE POLINT(XA,YA : GLNARRAY; N : INTEGER ;X : DOUBLE ; VAR Y,DY :DOUBLE);
VAR

```

```

NS,M,I : INTEGER;
W,HP,HO,DIFT,DIF,DEN : DOUBLE;
C,D : GLNARRAY;

```

```

BEGIN
  NS:=1;
  DIF:=ABS(X-XA[1]);
  FOR I:=1 TO N DO
    BEGIN
      DIFT:=ABS(X-XA[I]);
      IF (DIFT < DIF) THEN
        BEGIN
          NS:=I;
          DIF:=DIFT;
        END;
      C[I]:=YA[I];
      D[I]:=YA[I];
    END;
  Y:=YA[NS];
  NS:=NS-1;
  FOR M:=1 TO N-1 DO
    BEGIN
      FOR I:=1 TO N-M DO
        BEGIN
          HO:=XA[I]-X;
          HP:=XA[I+M]-X;
          W:=C[I+1]-D[I];
          DEN:=HO-HP;
          IF(DEN = 0) THEN
            II:=1;
          DEN:=W/DEN;
          D[I]:=HP*DEN;
          C[I]:=HO*DEN;
        END;
      IF (2*NS < N-M) THEN
        DY:=C[NS+1]
      ELSE
        BEGIN
          DY:=D[NS];
          NS:=NS-1;
        END;
      Y:=Y+DY;
    END
  END;
END;

```

{ integral along phi direction with fixed range in polar coordinate system }

```

PROCEDURE TRAPZD_PHI(Y,A,B : DOUBLE; VAR S : DOUBLE;
  N : INTEGER; VAR GLIT : INTEGER);
VAR

```

```

J : INTEGER;
X,TNM,SUM,DEL : DOUBLE;

BEGIN
IF (N = 1) THEN
  BEGIN
    S:=0.5*(B-A)*(FUNC(A,Y)+FUNC(B,Y));
    GLIT:=1;
  END
ELSE
  BEGIN
    TNM:=GLIT;
    DEL:=(B-A)/TNM;
    X:=A+0.5*DEL;
    SUM:=0.0;
    FOR J:=1 TO GLIT DO
      BEGIN
        SUM:=SUM+FUNC(X,Y);
        X:=X+DEL;
      END;
    S:=0.5*(S+(B-A)*SUM/TNM);
    GLIT:=2*GLIT;
  END
END;

```

(integral along phi direction with fixed range in polar coordinate system)

```

FUNCTION QROMB_PHI(Y,A,B : DOUBLE):DOUBLE;
LABEL 99;
CONST
  EPS = 1.0E-6;
  JMAX = 16;
  JMAXP = 17;
  K = 4;

VAR
  I,J,GLIT_PHI : INTEGER;
  DSS,SS : DOUBLE;
  H,S : ARRAY[1..JMAXP] OF DOUBLE;
  C,D : GLNARRAY;

BEGIN
  H[1]:=1.0;
  FOR J:=1 TO JMAX DO
    BEGIN
      TRAPZD_PHI(Y,A,B,S[J],J,GLIT_PHI);
      IF (J >= K) THEN
        BEGIN
          FOR I := 1 TO K DO
            BEGIN

```

```

        C[I] := H[J-K+1];
        D[I] := S[J-K+1]
    END;
    POLINT(C,D,K,0.0,SS,DSS);
    IF (ABS(DSS) < (EPS*ABS(SS))) THEN GOTO 99;
    END;
    S[J+1] := S[J];
    H[J+1] := 0.25*H[J];
    END;
    II := 2;
99: QROMB_PHI := SS;
END;

```

{ integral along range direction on a function whose points are phi_direction
integrals in polar coordinate system }

```

PROCEDURE TRAPZD_RHO(A,B : DOUBLE; VAR S : DOUBLE;
    N : INTEGER; VAR GLIT : INTEGER; CIRCLE : INTEGER);
VAR
    J : INTEGER;
    RHO,TNM,SUM,DEL,PHI2_A,PHI2_B,PHI2_RHO : DOUBLE;

```

```

BEGIN
    IF (N = 1) THEN
        BEGIN
            IF (CIRCLE <> 0) THEN      { integral on a semi_circle }
                BEGIN                { CIRCLE = 2 for left semi_circle }
                    PHI2_A := CIRCLE*PI/2;    { CIRCLE = 1 for right semi_circle }
                    PHI2_B := PHI2_A          { integral limits are PI/2 to PI or 0 to PI/2 }
                END
            ELSE
                BEGIN
                    PHI2_A := PHI_LIMIT(CENTER_Y, LONG_AXIS, SHORT_AXIS, A);
                    PHI2_B := PHI_LIMIT(CENTER_Y, LONG_AXIS, SHORT_AXIS, B)
                END;
                R := SQR(SQR(A)+SQR(HEIGHT));
                R4 := SQR(SQR(R));
                SIGMA := SIGMA0(ARCTAN(A/HEIGHT),PAR);
                S := A*QROMB_PHI(A,PHI1,PHI2_A)*SIGMA/R4;
                R := SQR(SQR(B)+SQR(HEIGHT));
                R4 := SQR(SQR(R));
                SIGMA := SIGMA0(ARCTAN(B/HEIGHT),PAR);
                S := S+B*QROMB_PHI(B,PHI1,PHI2_B)*SIGMA/R4;
                S := 0.5*(B-A)*S;
                GLIT := 1;
            END
        ELSE
            BEGIN
                TNM := GLIT;
                DEL := (B-A)/TNM;

```



```

RHO:=A+0.5*DEL;
SUM:=0.0;
FOR J:=1 TO GLIT DO
  BEGIN
    IF (CIRCLE <> 0 ) THEN
      BEGIN
        PHI2_RHO :=CIRCLE*PI/2;
      END
    ELSE
      BEGIN { integral limit on ellipse edge }
        PHI2_RHO := PHI_LIMIT(CENTER_Y, LONG_AXIS, SHORT_AXIS, RHO);
      END;
      R := SQRT(SQR(RHO)+SQR(HEIGHT)); { range }
      R4 := SQR(SQR(R));
      SIGMA := SIGMA0(ARCTAN(RHO/HEIGHT), PAR); { sigma0 }
      SUM:=SUM+RHO*QROMB_PHI(RHO, PHI1, PHI2_RHO)*SIGMA/R4;
      RHO:=RHO+DEL;
    END;
    S:=0.5*(S+(B-A)*SUM/TNM);
    GLIT:=2*GLIT;
  END
END;

```

```

PROCEDURE QROMB_RHO(A,B : DOUBLE; VAR SS : DOUBLE; CIRCLE :INTEGER);

```

```

  LABEL 99;

```

```

  CONST

```

```

    EPS = 1.0E-5;

```

```

    JMAX = 16;

```

```

    JMAXP = 17;

```

```

    K = 4;

```

```

  VAR

```

```

    I,J, GLIT_RHO : INTEGER;

```

```

    DSS : DOUBLE;

```

```

    H,S : ARRAY[1..JMAXP] OF DOUBLE;

```

```

    C,D : GLNARRAY;

```

```

  BEGIN

```

```

    H[1]:=1.0;

```

```

    FOR J:=1 TO JMAX DO

```

```

      BEGIN

```

```

        TRAPZD_RHO(A,B,S[J],J, GLIT_RHO, CIRCLE);

```

```

        IF (J >= K) THEN

```

```

          BEGIN

```

```

            FOR I := 1 TO K DO

```

```

              BEGIN

```

```

                C[I] := H[J-K+I];

```

```

                D[I] := S[J-K+I]

```

```

              END;

```

```

POLINT(C,D,K,0.0,SS,DSS);
IF (ABS(DSS) < (EPS*ABS(SS))) THEN GOTO 99;
END;
S[J+1]:=S[J];
H[J+1]:=0.25*H[J];
END;
II := 3;
99::
END;

```

```

BEGIN
ASSIGN(FDAT,'G5_S.DAT');
HALFBEAM_INT := 2*BEAMWIDTH_3DB;    { integral beam }
HALFBEAM_EFF := 1.2*BEAMWIDTH_3DB/2; { effective beam : illumination area }
S_PAR := 0.05*D_R;                  { exponential parameter step }
PAR := 0.8*D_R;                      { initial value of exponential para. }
WHILE (PAR <= (2.5*D_R)) DO
BEGIN
APPEND(FDAT);
WRITELN(FDAT);
WRITELN(FDAT,PAR/D_R);
CLOSE(FDAT);
THETA := 0;                          { initial incidence angle }
WHILE (THETA < 51) DO
BEGIN
II := 0;                            { II = 0 indicate finishing integral correctly }
R_THETA := THETA*D_R;
R0 := HEIGHT/COS(R_THETA);
Y0 := HEIGHT*TAN(R_THETA);
ELLIPSE(R_THETA,HALFBEAM_INT,CENTER_Y,LONG_AXIS,SHORT_AXIS);
S1_CIRCLE := 0;
S2_CIRCLE := 0;
SINT := 0;
CIRCLE := 0;
PHI1 := 0;
IF (CENTER_Y < LONG_AXIS) THEN
BEGIN
{ integral on left semi_circle }
CIRCLE := 2;
PHI1 := PI/2;                      { low integral limit for angle }
X1 := SHORT_AXIS*SQR(1-SQR(CENTER_Y/LONG_AXIS));
Y1 := LONG_AXIS-CENTER_Y;
IF (X1 > Y1) THEN X_Y := X1 { up integral limit for range }
ELSE X_Y := Y1;
QROMB_RHO(0.0,X_Y,S2_CIRCLE,CIRCLE);

{ integral on right semi_circle }
CIRCLE := 1;
PHI1 := 0;                          { low integral limit for angle }
QROMB_RHO(0.0,X1,S1_CIRCLE,CIRCLE);

```

```

CIRCLE := 0;
RHO1 := X1;           { low integral limit for range }
RHO2 := CENTER_Y+LONG_AXIS { up integral limit for range }
END
ELSE
BEGIN
  RHO1 := CENTER_Y-LONG_AXIS; { low integral limit for range }
  RHO2 := CENTER_Y+LONG_AXIS { up integral limit for range }
END;
IF (RHO2 > RHO1) THEN

  { integral on ellipse }
  QROMB_RHO(RHO1,RHO2,SINT,CIRCLE);
  SINT := SINT+S1_CIRCLE+S2_CIRCLE;
  ELLIPSE(R_THETA,HALFBEAM_EFF,CENTER_Y,LONG_AXIS,SHORT_AXIS);

  { calculate illuminated area }
  AILL := PI*LONG_AXIS*SHORT_AXIS;
  SINT := 2*SINT*SQR(SQR(R0))/AILL;
  WRITELN('PAR,THETA,SIGMA0,SS = ',PAR:10:5,THETA:10:5,SINT:10:7,
          SINT/SIGMA0(R_THETA,PAR):10:7);

  IF (II < 0) THEN
    WRITELN('ERROR IN ',PAR/D_R:10:5,' WITH ',II);
    APPEND(FDAT);
    WRITELN(FDAT,DB(SINT):10:6,DB(SINT/SIGMA0(R_THETA,PAR)):10:6);
    CLOSE(FDAT);
    FOR JJ := 1 TO 5 DO
      WRITELN;
      THETA := THETA + S_THETA
    END;
    PAR := PAR+S_PAR
  END
END.

```